UNIVERSITÉ DE MONTRÉAL

CENTRE DE RECHERCHE SUR LES TRANSPORTS

CASE POSTALE 6128, SUCCURSALE A, MONTRÉAL, QUÉBEC, H3C 3J7

MINIMUM COST FLOW PROBLEM WITH PIECEWISE NON-CONTINUOUS ARC COSTS

by

Jean Éthier* and Jacques A. Ferland**

* Centre de recherche sur les transports, Université de Montréal et Département de mathématiques appliquées, École Polytechnique.

** Département d'informatique et de recherche opérationnelle, Université de Montréal.

This research was supported by NSERC (A8312) and by FCAC (0899).

Cette étude a été publiée grâce à une subvention du fonds F.C.A.R. pour l'aide et le soutien à la recherche.

Université de Montréal

Centre de recherche sur les transports - Publication #345
Département d'informatique et
de recherche opérationnelle - Publication #508

Juin 1984

Dépôt légal - 1er trimestre 1988

ABSTRACT

In this paper the minimum cost flow problem with piecewise linear and non-continuous arc costs is analyzed. Two different approaches are examined: Lagrangian relaxation and Benders decomposition. Numerical results are included to compare the approaches.

Key words: minimum cost flow problem; Lagrangian relaxation; Benders decomposition; piecewise linear.

- - - - - - - - - - -

RÉSUMÉ

Dans cette publication nous étudierons le problème de déterminer un flot à coût minimum dans un réseau où les coûts du flot sur les arcs sont des fonctions linéaires par morceaux admettant des continuités. Deux approches différentes de résolution sont analysées : la première est basée sur la relaxation lagrangienne et la seconde sur la décomposition de Benders. Des résultats numériques sont inclus pour comparer les deux approches.

Mots clés : flot à coût minimum; relaxation lagrangienne; décomposition de Benders; linéaire par morceaux.

1. Introduction

In this paper the minimum cost flow problem with piecewise linear and non-continuous arc costs is analyzed. The problem is reformulated to replace each arc by many arcs, each of these having a linear unitary cost and a set-up cost. The problem then includes two types of variables: flow variables and decision variables associated with set-up costs.

Two different approaches to this problem are investigated. First a Lagrangian relaxation of the problem which is generated by dualizing the constraints linking the flow variables and the decision variables is introduced, and this relaxation is solved with a subgradient procedure. The second approach solves the problem with Benders decomposition.

In Section 2, the problem is specified and reformulated with decision variables. The first and second approaches are analyzed in Section 3 and 4, respectively. In Section 5, numerical results are given to compare these approaches.

2. Problem statement

Let $G = (N,A)$ be an oriented graph where N is the set of nodes and $A = \{(i,j) : i,j \in N\}$ is the set of arcs. The minimum cost flow problem is specified as follows:

$$\text{Min} \sum_{(i,j)\in A} C_{ij}(x_{ij})$$

$$\text{Subject to} \sum_{j\in Q_i} x_{ij} - \sum_{j\in R_i} x_{ji} = v_i \qquad i\in N \tag{2.1}$$

$$0 \le x_{ij} \le u_{ij} \qquad (i,j)\in A$$

where $Q_i = \{j \in N: (i,j) \in A\}$, $R_i = \{j \in N: (j,i) \in A\}$. If $v_i > 0$, then node i is a source, if $v_i < 0$, it is a sink, and $\sum_{i \in N} v_i = 0$. The cost function analyzed here takes the form $C_{ij}(0)=0$,

$$C_{ij}(x_{ij}) = \sum_{\ell=1}^{k-1} [d_{ij}^{\ell} + c_{ij}^{\ell} (w_{ij}^{\ell} - w_{ij}^{\ell-1})] + d_{ij}^{k} + c_{ij}^{k} (x_{ij} - w_{ij}^{k-1})$$

if $x_{ij} \in (w_{ij}^{k-1}, w_{ij}^{k}]$. The c_{ij}^{ℓ}, w_{ij}^{ℓ} and d_{ij}^{ℓ} are scalars, and we define K_{ij} as the value of k such that $w_{ij}^{k} = u_{ij}$. We also assume that $0 = w_{ij}^{0} < w_{ij}^{1}$, $< \ldots < w_{ij}^{K_{ij}} = u_{ij}$, and for all $1 \leq k \leq K_{ij}$

$$c_{ij}^{k-1} < c_{ij}^{k} \quad \text{and} \quad d_{ij}^{k} \geq 0$$

(2.2) or

$$c_{ij}^{k-1} \leq c_{ij}^{k} \quad \text{and} \quad d_{ij}^{k} > 0 \quad .$$

An objective function of this type is illustration in Figure 2.1

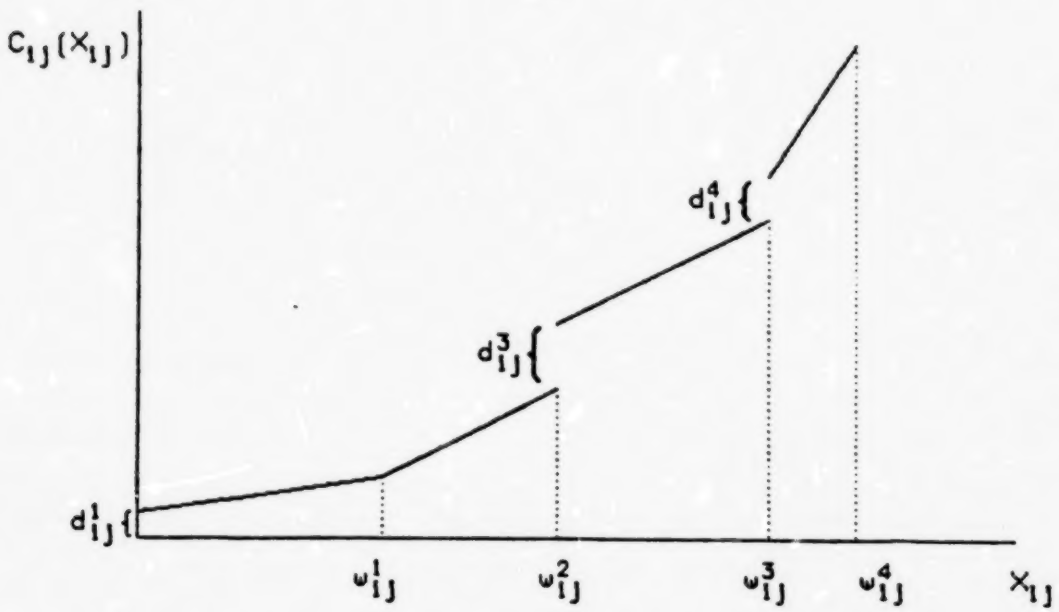

FIGURE 2.1

The objective function given above makes problem (2.1) difficult to solve. So we reformulate the problem by replacing each arc by K_{ij} arcs each of these having a unitary cost of c_{ij}^k, a set up cost of d_{ij}^k, and a capacity of $u_{ij}^k = w_{ij}^k - w_{ij}^{k-1}$. The flow on each arc is denoted x_{ij}^k. The new problem is specified as follows

$$\text{Min} \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^k + d_{ij}^k y_{ij}^k)$$

$$\text{Subject to} \sum_{j\in Q_i} \sum_{k=1}^{K_{ij}} x_{ij}^k - \sum_{j\in R_i} \sum_{k=1}^{K_{ji}} x_{ji}^k = v_i \qquad i\in N$$

$$0 \le x_{ij}^k \le u_{ij}^k \qquad (i,j)\in A\ ,\ 1 \le k \le K_{ij}$$

(2.3)
$$y_{ij}^k = 1 \iff x_{ij}^k > 0 \qquad (i,j)\in A\ ,\ 1 \le k \le K_{ij}$$

$$x_{ij}^k > 0 \Rightarrow x_{ij}^\ell = u_{ij}^\ell \qquad (i,j)\in A\ ,\ 1 \le k \le K_{ij},\ 1 \le \ell \le k-1$$

$$y_{ij}^k = 0 \text{ or } 1 \qquad (i,j)\in A\ ,\ 1 \le k \le K_{ij}$$

It is easy to verify that problem (2.3) and (2.1) are equivalent ([3, Proposition 1.3]). Unfortunatly problem (2.3) includes two sets of constraints involving conditional values of some variables in terms of the values of other variables. To remove this difficulty, consider the following problem:

$$\text{Min} \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^k + d_{ij}^k y_{ij}^k)$$

$$\text{Subject to} \sum_{j\in Q_i} \sum_{k=1}^{K_{ij}} x_{ij}^k - \sum_{j\in R_i} \sum_{k=1}^{K_{ji}} x_{ji}^k = v_i \qquad i\in N$$

$$(2.4) \qquad 0 \le x_{ij}^k \le u_{ij}^k \qquad (i,j) \in A\,, \quad 1 \le k \le K_{ij}$$

$$x_{ij}^k \le y_{ij}^k \, u_{ij}^k \qquad (i,j) \in A\,, \quad 1 \le k \le K_{ij}$$

$$y_{ij}^k \le y_{ij}^{k-1} \qquad (i,j) \in A\,, \quad 1 \le k \le K_{ij}$$

where $y_{ij}^0 \equiv 1$ for all $(i,j) \in A$.

Problems (2.4) and (2.1) are not equivalent, and in fact (2.4) is a relaxation of (2.1). On the other hand, referring to assumption (2.2) on the cost functions $C_{ij}(x_{ij})$, and using the following results, it is easy to generate an optimal solution for (2.1) with a value equal to the optimal value of (2.4). Proofs of the following Propositions are included in [3].

Proposition 2.1 [3, Proposition 1.9] Let x,y be an optimal solution of (2.4) which does not satisfy the constraints $x_{ij}^k > 0 \Rightarrow x_{ij}^\ell = u_{ij}^\ell$ $1 \le \ell \le k-1$ for some $(i,j) \in A$ and $1 \le k \le K_{ij}$. Let $t = \min_{1 \le k \le K_{ij}} \{k : x_{ij}^k < u_{ij}^k\}$ and $t^* = \max_{1 \le k \le K_{ij}} \{k : x_{ij}^k > 0\}$. Then $c_{ij}^t = c_{ij}^{t^*}$ and $x_{ij}^t + x_{ij}^{t^*} > u_{ij}^t$.

Proposition 2.2 [3, Proposition 1.8] Let x,y be an optimal solution of (2.4) satisfying all constraints of (2.1) except some constraints $y_{ij}^k = 1 \Rightarrow x_{ij}^k > 0$ for some $(i,j) \in A$, $1 \le k \le K_{ij}$. Let $k^* = \max_{1 \le k \le K_{ij}} \{k : x_{ij}^k > 0\}$ and $t^* = \max_{1 \le k \le K_{ij}} \{k : y_{ij}^k = 1\}$. Then $t^* > k^*$ and $d_{ij}^k = 0$ for $k^* < k \le t^*$.

Any solution of (2.4) satisfies the constraints

$$x_{ij}^k > 0 \Rightarrow y_{ij}^k = 1 \qquad (i,j)\in A,\ 1 \le k \le K_{ij} .$$

Proposition 2.1 and 2.2 indicate that under assumption (2.2) an optimal solution of (2.1) can be derived from an optimal solution of (2.4) by reallocating the flow appropriately among the arcs $(i,j)\in A$, $1 \le k \le K_{ij}$. Furthermore the value of the solution is equal to the optimal value of (2.4).

3. Subgradient approach

Problem (2.4) will be solved via Lagrangian relaxation using subgradients techniques. A Lagrangian dual of (2.4) is specified as follows:

$$(3.1) \qquad \underset{\lambda,\nu \ge 0}{\text{Max}} \quad F(\lambda,\nu)$$

where

$$F(\lambda,\nu) = \underset{\substack{x\in X\\ y\in Y}}{\text{Min}} \Big\{ \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^k + d_{ij}^k y_{ij}^k + \lambda_{ij}^k (x_{ij}^k - y_{ij}^k u_{ij}^k) + \nu_{ij}^k (y_{ij}^k - y_{ij}^{k-1})) \Big\}$$

$$(3.2) \quad X = \Big\{x : \sum_{J\in Q_i} \sum_{k=1}^{K_{ij}} x_{ij}^k - \sum_{J\in R_i} \sum_{k=1}^{K_{ji}} x_{ji}^k = v_i,\ i\in N, \text{ and } 0 \le x_{ij}^k \le u_{ij}^k,\ (i,j)\in A,\ 1 \le k \le K_{ij}\Big\}$$

$$(3.3) \quad Y = \{y : y_{ij}^k = 0 \text{ or } 1, \quad (i,j)\in A,\ 1 \le k \le K_{ij}\} .$$

The problem specifying $F(\lambda,\nu)$ is separable in x and y as follows

$$F(\lambda,\nu) = G(\lambda,\nu) + H(\lambda,\nu)$$

where

$$(3.4) \quad G(\lambda,\nu) = \underset{x\in X}{\text{Min}} \Big\{ \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k + \lambda_{ij}^k) x_{ij}^k \Big\}$$

$$(3.5) \quad H(\lambda,\nu) = \underset{y\in Y}{\text{Min}} \Big\{ \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (d_{ij}^k - \lambda_{ij}^k u_{ij}^k + \nu_{ij}^k - \nu_{ij}^{k+1}) y_{ij}^k \Big\}$$

with $\nu_{ij}^{K_{ij}+1} \equiv 0$. To evaluate $H(\lambda,\nu)$ it suffices to take

$$\bar{y}_{ij}^{k} = \begin{cases} 1 & \text{if } d_{ij}^{k} - \lambda_{ij}^{k} u_{ij}^{k} + \nu_{ij}^{k} - \nu_{ij}^{k+1} < 0 \\ 0 & \text{otherwise.} \end{cases}$$

To evaluate $G(\lambda,\nu)$, a linear minimum cost flow problem has to be solved. Furthemore, given $\bar{x} \in X$ and $\bar{y} \in Y$ the solutions of (3.4) and (3.5) respectively, it is easy to reallocate the flow and adjust the values of the y_{ij}^{k} to generate a feasible solution of (2.4). One way of doing this is as follows: for all $(i,j) \in A$, determine k^* such that

$$\sum_{\ell=1}^{k^*-1} u_{ij}^{\ell} < \sum_{k=1}^{K_{ij}} \bar{x}_{ij}^{k} \leq \sum_{\ell=1}^{k^*} u_{ij}^{\ell}$$

and then

$$(3.6)\quad x_{ij}^{k} = \begin{cases} u_{ij}^{k} & \text{if } 1 \leq k < k^* \\ \sum_{k=1}^{K_{ij}} \bar{x}_{ij}^{k} - \sum_{\ell=1}^{k^*-1} u_{ij}^{\ell} & \text{if } k = k^* \\ 0 & \text{if } k^* + 1 \leq k \leq K_{ij} \end{cases}$$

$$(3.7)\quad y_{ij}^{k} = 1 \quad \text{if } x_{ij}^{k} > 0 \qquad 1 \leq k \leq K_{ij}\ .$$

The approach taken involves solving the dual (3.1) using subgradient techniques, and for each feasible solution λ , $\nu \geq 0$ of the dual, generating a feasible solution x,y of (2.4). It follows from weak Lagrangean duality that

$$F(\lambda,\nu) \leq V^* \leq \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^{k} x_{ij}^{k} + d_{ij}^{k} y_{ij}^{k})$$

(V^* ≡ optimal value of (2.4)). During the iterations of the procedure,

the value of the best feasible solution $\lambda^0 \geq 0$, $\nu^0 \geq 0$ of (3.1) (denoted by LB) and the value of the best feasible solution x^0, y^0 generated for (2.4) (denoted by UB) are stored, and the procedure stops whenever

$$UB - LB < \varepsilon$$

Then x^0, y^0 is ε-optimal for (2.4).

It is easy to verify that $F(\lambda, \nu)$ is concave in λ and ν and that given $\bar{x} \in X$ and $\bar{y} \in Y$ solutions of (3.4) and (3.5), respectively, a subgradient of F at (λ, ν) is the vector (ρ, Π) where

$$\rho_{ij}^k = \bar{x}_{ij}^k - \bar{y}_{ij}^k \, u_{ij}^k$$

$$\Pi_{ij}^k = \bar{y}_{ij}^k - \bar{y}_{ij}^{k-1}$$

for all $(i,j) \in A$, $1 \leq k \leq K_{ij}$. The algorithm used to solve the dual problem (3.1) is a variant of Wolfe's subgradient technique [6] as proposed by Carraresi and Gallo in [2]. It can be summarized as follows:

Step 0. Select an initial feasible solution λ^0, ν^0 for (3.1). Let (ρ^0, Π^0) be a subgradient of $F(\lambda, \nu)$ at (λ^0, ν^0). To evaluate $F(\lambda^0, \nu^0)$, optimal solutions $\bar{x}^0 \in X$ and $\bar{y}^0 \in y$ of (3.4) and (3.5), respectively, are determined. Let x^0, y^0 be a feasible solution of (2.4) associated with $\bar{x}^0$, $\bar{y}^0$.

Let

$$UB = \sum_{(i,j) \in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k \, x_{ij}^{0k} + d_{ij}^k \, y_{ij}^{0k}) \, , \quad LB = F(\lambda^0, \nu^0), \quad SG = \{(\rho^0, \Pi^0)\} \, ,$$

and NBG=0.

Step 1. (Modification Direction).

$$\text{Let } \bar{e}_{ij} = \sum_{\ell=1}^{|SG|} \frac{\rho_{ij}^\ell}{\|(\rho^\ell, \Pi^\ell)\|}. \quad \Delta$$

$$\bar{f}_{ij} = \sum_{\ell=1}^{|SG|} \frac{\Pi_{ij}^{\ell}}{\|(\rho^{\ell}, \Pi^{\ell})\| \cdot \Delta}$$

where $$\Delta = \sum_{\ell=1}^{|SG|} \frac{1}{\|(\rho^{\ell}, \Pi^{\ell})\|} .$$

Project this direction onto the feasible domain of (3.1) as follows

$$e_{ij} = \begin{cases} \bar{e}_{ij} & \text{if } \lambda_{ij}^{k} > 0 \text{ or if } \bar{e}_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

$$f_{ij} = \begin{cases} \bar{f}_{ij} & \text{if } \nu_{ij}^{k} > 0 \text{ or if } \bar{f}_{ij} \geq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Step 2. (Stopping criterion)

If $\|(e,f)\| > \varepsilon_1$, go to Step 3. If NBG $\leq \varepsilon_2$, then terminate the procedure. Otherwise, let (ρ, Π) be a subgradient of $F(\lambda,\nu)$ at (λ^p, ν^p). Then SG = $\{(\rho,\Pi)\}$, NBG = 0, and repeat Step 1.

Step 3. (Line search).

Complete a line search in the direction (e,f) specified in Step 1 using the procedure proposed by Carraresi and Gallo [2, Appendix]. Let $(\lambda^{p+1}, \nu^{p+1})$ be the optimal solution of (3.1) in the direction (e,f). The procedure generates also at least one new subgradient of F at $(\lambda^{p+1}, \nu^{p+1})$ if $(\lambda^{p+1}, \nu^{p+1}) = (\lambda^p, \nu^p)$ and two subgradients of F at $(\lambda^{p+1}, \nu^{p+1})$ if $(\lambda^{p+1}, \nu^{p+1}) \neq (\lambda^p, \nu^p)$. Introduce these subgradients in SG, and determine a feasible solution x^{p+1}, y^{p+1} of (2.4). Let NBG = NBG + $\|(\lambda^{p+1}, \nu^{p+1}) - (\lambda^p, \nu^p)\|$.

If $F(\lambda^{p+1}, \nu^{p+1}) > LB$, then $LB = F(\lambda^{p+1}, \nu^{p+1})$ and $(\lambda^0, \nu^0) = (\lambda^{p+1}, \nu^{p+1})$.

If $\sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^{p+1k} + d_{ij}^k y_{ij}^{p+1k}) < UB$, then

$UB = \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^{p+1k} + d_{ij}^k y_{ij}^{p+1k})$, and $(x^0, y^0) = (x^{p+1}, y^{p+1})$.

Let $p = p+1$ and repeat Step 1. □

It should be noted that, in our tests, the new subgradient of F at $(\lambda^{p+1}, \nu^{p+1})$ generated in Step 3 may be the same as one of the subgradients of F at (λ^p, ν^p) if the direction is not an ascent direction. In fact, the initial solution (λ^0, ν^0) was selected as follows. First a linear minimum cost flow problem was solved with arc costs

$$c_{ij}^k = \sum_{\ell=1}^{K_{ij}} (d_{ij}^\ell + c_{ij}^\ell u_{ij}^\ell)/u_{ij}^k \qquad (i,j)\in A, \quad 1 \le k \le K_{ij}.$$

Given an optimal solution $\bar{x}$ for this problem, a feasible solution (x,y) for (2.4) was generated according to relations (3.6) and (3.7). Given this solution (x,y), λ^0 and ν^0 are specified to maximize $F(\lambda,\nu)$.

This is a dual approach since we solve the dual (3.1) instead of solving the problem (2.1). However, given a feasible solution of the dual, it is easy to generate a feasible solution of the primal problem (2.1), but in general, the value of the primal objective function is not monotonitically decreasing. Moreover, since Y is not a convex set, only weak duality can be applied, and there could be a gap between the optimal values of the primal (2.4) and the dual (3.1). This is the major drawback of this approach.

4. Benders Decomposition

Problem (2.4) can be formulated as

$$\text{Min} \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^k + d_{ij}^k y_{ij}^k)$$

(4.1)

$$\text{Subject to} \quad 0 \le x_{ij}^k \le y_{ij}^k u_{ij}^k \qquad (i,j)\in A, \quad 1 \le k \le K_{ij}$$

$$x \in X, \; y \in \bar{Y}$$

where X is specified in (3.2) $\bar{Y} = \{y : y_{ij}^k \le y_{ij}^{k-1}, \; y_{ij}^k = 0 \text{ or } 1, \; y_{ij}^0 = 1, \; (i,j)\in A, \; 1 \le k \le K_{ij}\}$. This problem can be solved iteratively by Benders decomposition [1].

During the ℓth iteration, the master problem has the form

$$\text{Min} \quad z_0 + \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^k y_{ij}^k$$

$$\text{(4.2) Subject to} \quad \sum_{i\in N} v_i \alpha_i^{\xi} - \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} \lambda_{ij}^{\xi k} y_{ij}^k u_{ij}^k \le z_0 \qquad 1 \le \xi \le \bar{\xi}$$

$$\sum_{i\in N} v_i \alpha_i^{\tau} - \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} \lambda_{ij}^{\tau k} y_{ij}^k u_{ij}^k \le 0 \qquad 1 \le \tau \le \bar{\tau}$$

$$y \in \bar{Y}.$$

Let z_0^{ℓ}, y^{ℓ} be an optimal solution. With $y = y^{\ell}$, (4.1) reduces to a linear minimum cost flow problem.

$$\text{Min} \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} c_{ij}^k x_{ij}^k$$

$$\text{(4.3) Subject to} \quad 0 \le x_{ij}^k \le y_{ij}^{\ell k} u_{ij}^k \qquad (i,j)\in A, \quad 1 \le k \le K_{ij}$$

$$x \in X \; .$$

If x^{ℓ} is an optimal solution of (4.3), let λ^{ℓ} be a vector of optimal multipliers associated with constraints $x_{ij}^{k} \leq y_{ij}^{\ell k}\ u_{ij}^{k}$, and α^{ℓ} be a vector of optimal multipliers associated with constraints

$\sum_{j\in Q_i} \sum_{k=1}^{K_{ij}} x_{ij}^{k} - \sum_{j\in R_i} \sum_{k=1}^{K_{ji}} x_{ji}^{k} = v_i$. Introduce a new constraint in the master problem (4.2) (i.e. $\bar{\xi} = \bar{\xi} + 1$) specified as follows.

$$(4.4) \qquad \sum_{i\in N} v_i \alpha_i^{\ell} - \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} \lambda_{ij}^{\ell k}\ y_{ij}^{k}\ u_{ij}^{k} \leq z_0.$$

If (4.3) is infeasible, generate multipliers λ^{ℓ} et α^{ℓ} corresponding to the extreme ray for the dual of (4.2). The associated constraint to be introduced into the master problem (4.2) (i.e. $\bar{\tau} = \bar{\tau} + 1$) is of the form

$$(4.5) \qquad \sum_{i\in N} v_i \alpha_i^{\ell} - \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} \lambda_{ij}^{\ell k}\ y_{ij}^{k}\ u_{ij}^{k} \leq 0.$$

Notice that these multipliers are easily obtained with RNET [4] used to solved (4.3). Indeed, RNET introduces artificial arcs in such a way that the resulting extended problem corresponding to (4.3) is always feasible. If in fact (4.3) is not feasible without these artificial arcs, then the multipliers required to generate (4.5) are obtained from the optimal multipliers for the extended problem.

It is well known that (4.2) is a relaxation of (4.1) and hence $z_0^{\ell} + \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^{k}\ y_{ij}^{\ell k}$ is a lower bound LB of the optimal value V* of (4.1). Furthermore, if x^{ℓ} is an optimal solution of (4.3), then (x^{ℓ}, y^{ℓ}) is a feasible solution of (4.1) and $\sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^{k}\ x_{ij}^{\ell k} + d_{ij}^{k}\ y_{ij}^{\ell k})$ is an upperbound of V*.

To initiate the algorithm, initial values of y must be selected. Of course, a first master problem with $\bar{\xi} = \bar{\tau} = 0$ can be considered, and an optimal solution is then $y'=0$. But to increase the chance of finding a feasible solution for (4.3) when $y=y'$, some minimum capacity constraints should be added to (4.2) with $\bar{\xi} = \bar{\tau} = 0$:

$$(4.6) \qquad \sum_{j \in Q_i} \sum_{k=1}^{K_{ij}} u_{ij}^k \, y_{ij}^k \geq v_i \quad \text{if} \quad v_i > 0$$

$$(4.7) \qquad \sum_{j \in R_i} \sum_{k=1}^{K_{ji}} u_{ji}^k \, y_{ji}^k \geq -v_i \quad \text{if} \quad v_i < 0 \ .$$

These constraints require that the total capacity of the incident arcs to a given node satisfy the demand at this node.

Problem (4.2) is a 0-1 linear programming problem and as such it is not easy to solve. Two different approaches were analysed. In the first approach (4.2) was solved exactly using a branch and bound technique. The second approach involved a heuristic method. Two different heuristics were considered.

First, a heuristic method for handling (4.2) with $\bar{\xi} = 1$ or $\bar{\tau} = 1$ was developed. Then each time a new constraint was generated, a linear combination of the constraints was composed to replace (4.2) by an approximation where $\bar{\xi} = 1$ or $\bar{\tau} = 1$. However, even if a larger weights were allowed for more recently generated constraints, the approximation was not very efficient and the results were not encouraging.

The second heuristic generates a feasible solution of (4.2) by rounding up a continuous solution for the relaxed version of (4.2) where the constraints $y_{ik}^k = 0$ or 1 are replaced by $y_{ik}^k \geq 0$. Again the results were not encouraging.

Finally, the approach adopted is a compromise involving the use of the second heuristic and solving (4.2) exactly with branch-and-bound when the solution obtained with the heuristic has a value exceeding the upperbound.

The algorithm is summarized as follows

Step 0. Generate $y' \in Y$ by solving (4.2) with $\bar{\xi} = \bar{\tau} = 0$ and including constraints (4.6) and (4.7). Let $LB = \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^k y_{ij}'^k$, $UB = \infty$,

Step 1. Solve the relaxed master problem (4.2). Let z_0^ℓ, y^ℓ be a feasible solution generated with the second heuristic. If $z_0^\ell + \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^k y_{ij}^{\ell k} > UB$, move to Step 2. Otherwise,

$LB = \text{Max}\{LB, z_0^\ell + \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^k y_{ij}^{\ell k}\}$ and move to Step 3.

Step 2. Solve (4.2) with a branch and bound procedure. Let z_0^ℓ, y^ℓ be an optimal solution, $LB = \text{Max}\{LB, z_0^\ell + \sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} d_{ij}^k y_{ij}^{\ell k}\}$. If $UB-LB<\varepsilon$, terminate with x^0, y^0 an ε-optimal solution.

Step 3. Solve (4.3) with $y=y^\ell$.

If (4.3) is infeasible, generate a new constraint of the type (4.5) to be introduced into the relaxed master problem. $\bar{\tau} = \bar{\tau} + 1$, $\ell = \ell+1$ and repeat Step 1.

Otherwise, let x^ℓ an optimal solution of (4.3). If

$\sum_{(i,j)\in A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^{\ell k} + d_{ij}^k y_{ij}^{\ell k}) < UB$, then $UB = \sum_{(i,j)\ A} \sum_{k=1}^{K_{ij}} (c_{ij}^k x_{ij}^{\ell k} + d_{ij}^k y_{ij}^{\ell k})$, $x^0 = x^\ell$ and $y^0 = y^\ell$. If $UB - LB < \varepsilon$, terminate with x^0, y^0 an ε-optimal solution. Otherwise, generate a new constraint of type (4.4) to be introduced into the relaxed master problem. $\bar{\xi} = \bar{\xi} + 1$, $\ell = \ell + 1$ and report Step 1.

5. Numerical results

Both algorithms were tested on 15 randomly generated problems. The smallest problem had 5 nodes and 30 arcs; the largest one had 100 nodes and 500 arcs. Tables 5.1 and 5.2 give the detailed results including the average of the number of iterations, the gap (in %) before, and after and the execution time in seconds.

From table 5.1 it appears that the subgradient approach is very fast and could handle large problems. However, we see from the gaps (before and after) that the quality of the solution was not improved significantly in most of the cases (the average improvement is 7.63 - 6.95 = .68%). It appears also that good primal solutions are generated rather randomly. So the quality of the primal solution is more or less proportionnal to the number of iterations. This is a limitation of the algorithm since the modification direction is given by the subgradient which does not necessarely give the best direction; in many cases, the algorithm is stopped prematurely.

The Benders decomposition gives some good results with small problems (see table 5.2). The optimal solution was found for 7 problems (out of 15). The algorithm can quite easily handle problems with 150 arcs and 20 nodes or less. Over 150 arcs and 20 nodes and up to 500 arcs the results are acceptable. Over 500 arcs, the algorithm is not efficient because of time and memory requirements. Note that even when the solution of 4.3 gives the optimal solution (when the upper bound is set to the optimal value) it can take 3 to 4 iterations to bring the lower bound to this value and at each of these steps, a branch-and-bound is necessarily to solve (4.2). This increases significantly the execution time of the algorithm.

In summary, the subgradient approach is fast but the quality of the solution found is somewhat impredictable and the resulting gap may not be very significant. The Benders decomposition approach gives very good results on problems with 150 arcs or less. Above this figure the results are variable, and at over 500 arcs the algorithm is not efficient.

Problem	Number of iteration	gap (%) before	gap (%) after	Execution time (sec)
1	500	3.66	2.27	59.35
2	317	6.98	5.44	35.85
3	500	3.81	2.35	28.79
4	11	7.93	7.93	1.16
5	408	7.30	2,58	31.00
6	2	4.68	4.68	.92
7	215	2.68	2.56	77.16
8	49	6.07	5.93	20.09
9	249	16.29	14.98	56.04
10	265	2.24	2.23	85.04
11	236	1.05	1.05	22.58
12	26	.12	.12	2.90
13	9	2.00	1.75	1.71
14	14	3.14	3.14	2.29
15	9	46.79	46.79	2.16
Average	181.78	7.63	6.95	28.43

Table 5.1 (Subgradient approach)

Problem	Number of iteration	gap (%) before	gap (%) after	Execution time (sec)
1	13	8.70	0.00	35.37
2	31	87.01	0.00	180.49
3	1	26.26	0.00	.77
4	27	0.00	0.00	321.24
5	11	4.42	0.00	80.54
6	7	6.45	0.00	84.38
7	8	40.30	38.86	>1000
8	3	39.42	38.78	40.00
9	9	7.10	6.28	>1000
10	2	62.41	62.30	15.06
11	50	1.07	.51	889.39
12	26	.46	0.00	686.10
13	13	2.68	1.84	>1000
14	44	64.17	16.94	>1000
15	5	46.05	45.01	50.00
Average	20.00	20.71	5.37	523.19

Table 5.2 (Benders decomposition approach)

BIBLIOGRAPHIE

[1] J.F. Benders, "Partitioning procedures for solving mixed variables programming problems", Numerische Mathematik 4 (1962) 238-252.

[2] C. Carraresi, G. Gallo, "Optimal location of files and programs in computer networks", Mathematical Programming Study 20 (1982) 39-53.

[3] J. Ethier, "Problème de flot à fonction objectif linéaire par morceaux et non-continue", Mémoire de maîtrise, Département d'informatique et de recherche opérationnelle, Université de Montréal, Canada (1984).

[4] M. Grigoriadis, T. Hsu, "The Rutgers minimum cost network flow subroutine", Sigmap 26 (1979) 17-18.

[5] A.H. Land, S. Powell, Fortran codes for mathematical programming (Wiley 1973).

[6] P. Wolfe, "A method of conjugate subgradients for minimizing non-differential functions", Mathematical Programming Study 3 (1975) 145-173.

[7] P. Wolfe, "Finding the nearest point in a polytope", Mathematical Programming Study 11 (1976) 128-149.